

Presented by Team 900  
The Zebracorns



# Strategy in the Design Process

---

Inspired by Team 1114 Simbotics presentation  
by Karthik Kanagasabapathy

[https://www.youtube.com/watch?v=Apk\\_X-maRf8](https://www.youtube.com/watch?v=Apk_X-maRf8)

# What is a Strategy?



## Dictionary

- Strategy – a plan of action or policy designed to achieve a major or overall aim.

## In this presentation

- A plan for how a robot or alliance should behave on the field.

# Strategy Design:

designing a strategy for the game before designing a robot.

What is “designing a strategy?”

- Figuring out what type of gameplay you can use to most likely be in the winning alliance at competition.
- Big Questions:
  - What will the winning alliances at regionals be doing?
    - How can we best fit into those types of alliances?

# Why do Strategy Design?

- It is very hard to create a competitive robot without having the right gameplay objectives in mind.
  - It is easy to get wrapped up in cool mechanism ideas, and then miss out on strategically superior alternatives.

Gameplay at competition often converges to fit a certain set of strategies

- Determining these strategies ahead of time (and which you can best build for) will mean you can make more deliberate and effective design decisions.



# Team 900's Crystal Ball Team



The Crystal Ball Team's job is strategy brainstorming and analysis.

The team is a select group of students who have been training beforehand (analyzing past games, etc.)

- The group is kept small (3 to 10 students) mainly to streamline the analysis process.
  - Large group discussions often move *very* slowly and in circles.
- During this time, the rest of the team is also discussing rules, and are free to stop in and pitch ideas to the main analysis team.

# Strategy Design Ground Rules

## Avoid discussing robot designs

- Students will often get wrapped up in robot / mechanism ideas instead of proper analysis. This can be very distracting.
- Only discuss robot design during final debates, and only to the extent where you decide if a strategy can be built for.
  - Example: Ultimate Ascent 30 point climb

## No possessive pronouns

- Do not label things “his idea” or “my idea” – this makes debates personal and overly dramatic and can make students become overly possessive of ideas.

# Steps in Strategy Design

The first three days of build is a reasonable amount of time to devote towards Strategy Design.

## ----Day One----

- 1. Read the Rules
- 2. Brainstorm Strategies

## ----Day Two----

- 3. Analyze Strategies
- 4. Eliminate Strategies

## ----Day Three----

- 5. Debate / Select final strategy
- 6. Prioritize robot functions

# Day One

1. Reading the rules
2. Brainstorming Strategies



# Step #1: Reading the Rules

- Every student should read the manual, **at least 20 million times**
  - Students take note of odd rules – anything remotely interesting. Interesting rules lead to interesting strategies.
    - Example – Aerial Assist human player / truss rules
- Having students read on their own first is helpful
  - This helps every student develop their own unique perspective of the game. Divergent perspectives lead to more creative/unique ideas.

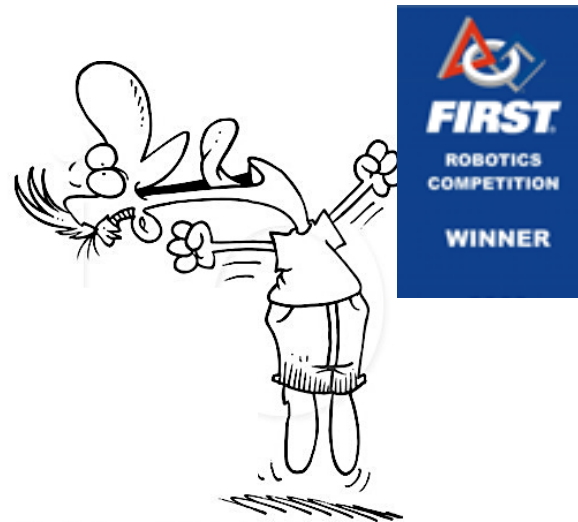
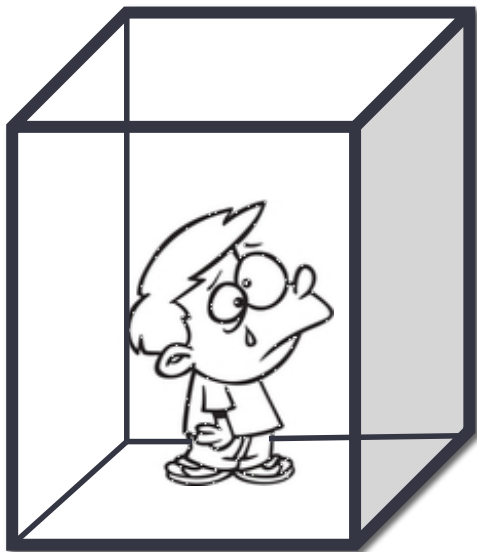
## Step #2: Brainstorming Strategies - Be Creative!

Think of as many strategies as possible.

- You should come up with 20+ ideas

Why we do this: The best strategy is never obvious.

- Look at every possible way of scoring, moving the robot on the field, and positioning it. (2013 – attaching to pyramid, 2012 – base of hoops)
- Look at how you can hinder opponents. (2002 – stealing the goals)



# Day Two

3. Analyzing Strategies

4. Eliminating Strategies



# Step #3: Analysis



Math == 😊

Find a way to make different tasks mathematically comparable (example: 2013 - 2pt and 3pt goals)

- This often includes breaking down cycles into different portions and assigning each part time ranges / success rates.

Simple, elegant analysis is *great*

- Be wary of getting into overly complicated math -- Too many variables can easily equate to a big margin for error and lots of headaches. (example – using “robot speed” as a variable)
- That being said, try to take as much into account as you can, or at least know what assumptions you are making



# 2014 - Simple Analysis

		Number of Cycles							
Points per Cycle		1	2	3	4	5	6	7	8
	1	1	2	3	4	5	6	7	8
	10	10	20	30	40	50	60	70	80
	11	11	22	33	44	55	66	77	88
	20	20	40	60	80	100	120	140	160
	21	21	42	63	84	105	126	147	168
	30	30	60	90	120	150	180	210	240
	31	31	62	93	124	155	186	217	248
	40	40	80	120	160	200	240	280	320
	41	41	82	123	164	205	246	287	328
	50	50	100	150	200	250	300	350	400
	51	51	102	153	204	255	306	357	408
	60	60	120	180	240	300	360	420	480

# 2014 - Simple Analysis

		Number of Cycles							
		1	2	3	4	5	6	7	8
Points per Cycle	1								
	10								
	11								
	20	20	40	60	80	100			
	21	21	42	63	84	105			
	30	30	60	90	120	150			
	31	31	62	93	124				
	40	40	80	120	160				
	41	41	82	123	164				
	50	50	100	150	200				
	51	51	102	153	204				
	60	60	120	180	240				

# Step #4: Eliminating strategies

Remove strategies that are clearly inferior to others, as can be seen through basic math/reasoning.

- Example: Triple Play low goal

In the end, 3 to 5 strategies should remain.

- These are strategies which are all competitive and out of which the “best” strategy is hard to definitively determine.
  - Usually the factors that might make each succeed over the others are difficult to quantify (presence of certain partners, difficulty of making mechanisms, abundance of defense, changes in the rules, etc.)
- The choice of the “best” strategy for your team will require a full debate.

# Day Three

5. Debate / Choose Final Strategy

6. Prioritize Robot Functions



# Step #5: Select Final Strategy

A team debate is a great way to do this

- This brings your entire team into the decision making process.

We have each strategy presented / advocated for by a strategy team member.

- This means that the students doing the main debating are already well informed of the pros/cons of each idea.

The team at large then asks questions/raises concerns

Finally, a vote is held.

- If your analysis and debating is done well, there should be an overwhelming consensus.

# Step #6: Prioritization

List out functions for your robot into the following categories.

- Robot must do \_\_\_\_\_ in order to play the game
- Robot must do \_\_\_\_\_ in order to perform our strategy
- It would be nice if the robot did \_\_\_\_\_ to help it play our strategy
- Other things we might want

This list will help guide your prototyping and design decisions

# Step #6: Prioritization - Example

Robot must do \_\_\_\_\_ in order to play the game

- Drive
- Comply with the rules

Robot must be able to do \_\_\_\_\_ in order to perform our strategy

- Shoot ball with x accuracy into x goal from x part of the field.
- Get between x part of the field and x loading station quickly regardless of defense
- Score x game pieces in autonomous into x goal
- Shoot over defense bots of x height
- Load from x loading station

It would be nice if the robot did \_\_\_\_\_ to help it play our strategy

- Vision systems
- Lock drivetrain when in place

Other things we might want

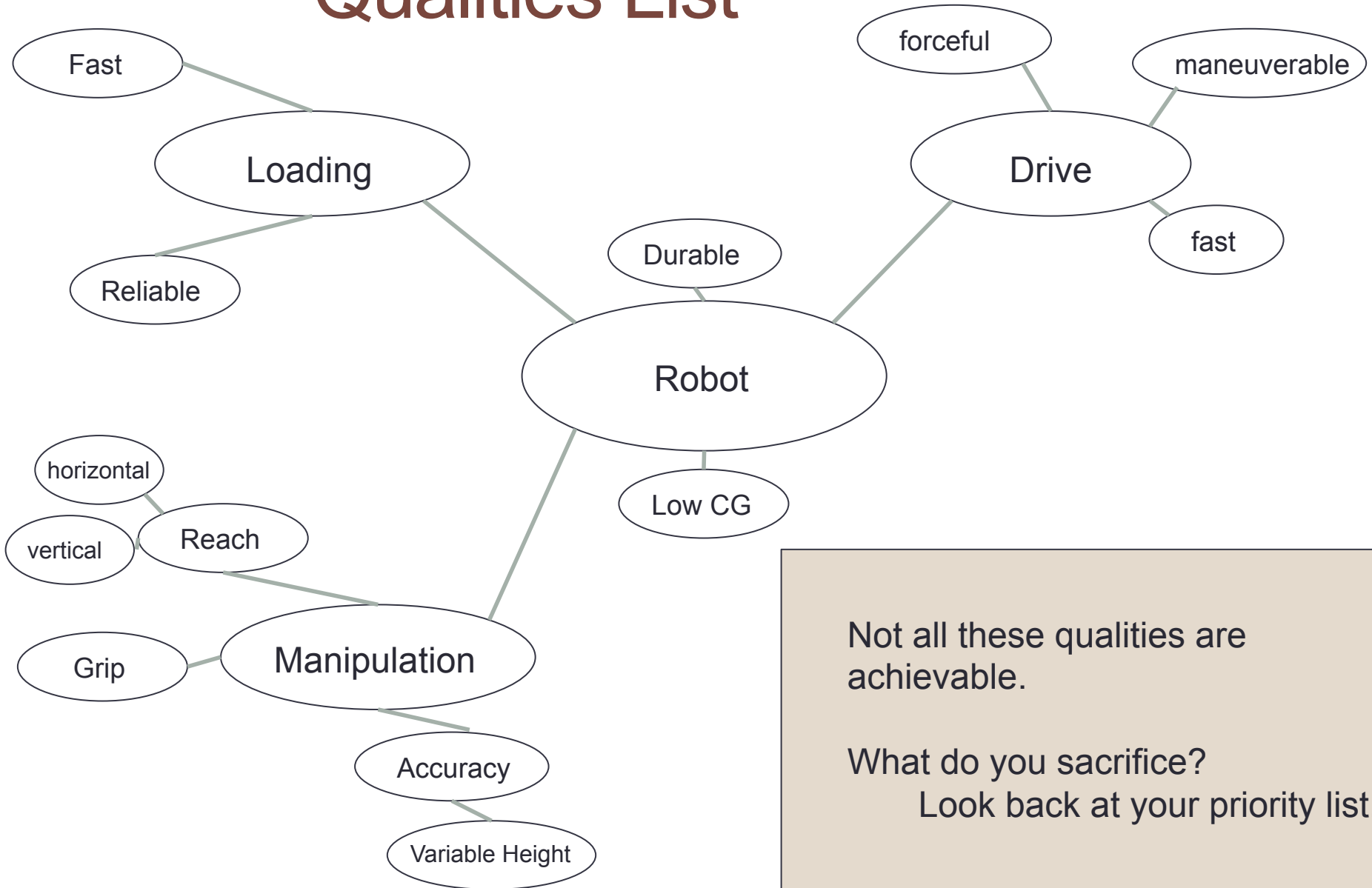
- The ability to pick up game pieces off the ground

# Qualities List





# Qualities List



# Other Things to be Aware of when Considering Strategies

Keeping Things Simple

Taking Defense Into Account

Other Teams at Competition

Resources

# Keeping it Simple

Simple objectives are *great*

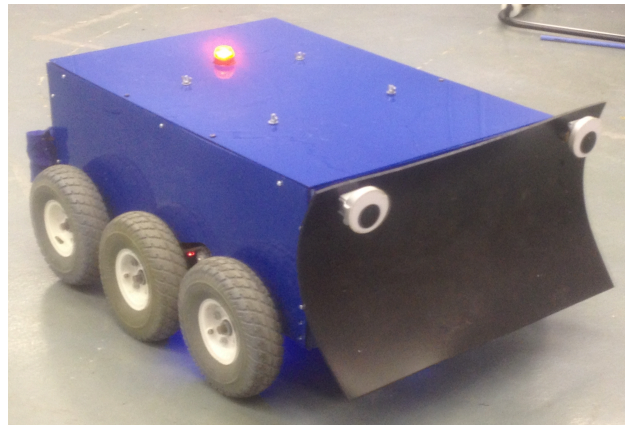
- This means less to build for; you can focus more attention on doing fewer things well
  - Your driver will be better practiced
  - You will likely have more robust systems

“...But being flexible will make us a great competitor.”

- This is *very* hard to do well. Teams who try to do everything on the field often lose focus on the tasks that are most valuable, and the quality of their gameplay suffers.
- Be aware of when your additions are taking away from your main functions.
  - This includes time – a new mechanism takes time to build, time to program, and time to practice with.
- That being said, it can be done. Just be careful and know your limits.

# Defense

- Defense is the death of great robots.
- There will be defensive robots. You will be fought against. Hard. *Especially* in eliminations.
- Always think “how can the other teams counter this strategy?”. Prepare for it, because it will happen.



# Other Bots

What other types of robot will be at competition?

- Realize what allies you will likely be working with.
  - Robot In 3 Days – gives a good idea of what teams will building.
- You'll have a better chance of being dominant doing something unique, rather than doing exactly what 30+ other robots will be built to do.



# In Summary

- Strategy design comes before robot design.
- Strategy design is important and effective, and should take about three days during build.
- Simple is effective – simple analysis and simple strategies do well.
- Know exactly how far you can stretch your resources
- Be creative and consider everything!

# Resources

- Effective FIRST Strategy – presentation by Karthik  
[https://www.youtube.com/watch?v=Apk\\_X-maRf8](https://www.youtube.com/watch?v=Apk_X-maRf8)
- ChiefDelphi – The robotics forum  
<http://www.chiefdelphi.com/forums/portal.php>
- Looking Forward FRC – videos describing important strategic elements in last year's game  
<https://www.youtube.com/channel/UCJwJT1XHnPw7ty1x8a3POzg/videos>
- FRC GameSense – mentors discussing strategies and gameplay  
<https://www.youtube.com/user/FRCGameSense>
- The Blue Alliance – video and results of past competitions  
<http://www.thebluealliance.com>

EWCP's Twenty Four – lots of lovely math  
<http://www.twentyfour.ewcp.org/>